Subassembly Detection and Optimal Assembly Sequence Generation through Elephant Search Algorithm

M. V. A. Raju Bahubalendruni
Department of Mechanical Engineering
National Institute of Technology Puducherry, Karaikal, Pincode-609609, India
Corresponding author: bahubalindruni@gmail.com

U. Sudhakar
Department of Mechanical Engineering
G. M. R. Institute of Technology, Rajam, Pincode -532127, India
E-mail: mr.u.sudhakar@gmail.com

K. V. Vara Lakshmi
Department of Mechanical Engineering
G. V. P. College of Engineering (A), Visakhapatnam, Pincode- 530048 India
E-mail: kvvaralakshmi@gvpce.ac.in

Abstract
Most of the engineering products are made with multiple components. The products with multiple subassemblies offer greater flexibility for parallel assembly operation and also disassembly operation during its end of life. Assembly cost and time can be minimized by reducing the number of assembly levels. In this paper Elephant search algorithm is used to perform Optimal Assembly Sequence Planning (OASP) in order to minimize the number of assembly levels. Subassembly identification technique is used as an integral part of algorithm to identify the parallel assembly possibilities. The proposed method is implemented on industrial products and a detailed comparative assessment has been made with suitable product illustrations to corroborate the efficiency.

Keywords- Assembly sequence planning, Elephant search algorithm, Swarm intelligence.

1. Introduction
There is an ever increasing demand for producing new products and supplying into market on-time towards fulfilling the customer needs, which made the industries to look towards new fabrication techniques and assembling methods. Product assembling is the major time consuming and expensive process for products with large part counts thus an industrial engineer always tries to reduce the overall fabrication cost and time of products (Bahubalendruni and Biswal, 2016a).

There exist several assembly sequence generation methods intended for different uses, the graphical based methods generally result in a single/multiple feasible assembly sequence (Bahubalendruni et al., 2015a). Some traditional methods and recent computational approaches offer the same by considering the assembly attributes as input (Vigano and Osorio Gomez, 2012; Bahubalendruni and Biswal, 2016b). However, these methods do not assure any optimal assembly sequences, an optimal assembly sequence is a feasible and stable assembly sequence which requires minimum assembly efforts and time (Deepak et al., 2018). Finding an optimal assembly sequence ensuing multiple assembly constraints including clash information is an extremely challenging task (Gunji et al., 2019). Several researchers implemented numerous optimization algorithms to solve the

assembly sequence planning at the price of quality of solution The researcher has ignored some of the assembly predicates to generate solution at faster rate but the quality of solution is not practically possible (Bahubalendruni et al., 2015b).

The journey of Artificial Intelligence (AI) based Optimal Assembly Sequence Planning (OASP) started with Genetic Algorithms (GA) and Simulated Annealing (SA) in the early 90's. The traditional algorithms have major limitation of premature convergence and often not suited for combinatorial discrete optimization problems (Deepak et al., 2018). Sometimes it demands a feasible solution to compute initial fitness value (Deepak et al., 2018).

The recent optimization techniques have been implemented to solve large products (Smith et al, 2001; Chen et al., 2010; Ghandi and Masehian, 2015; Bahubalendruni et al., 2016; Gunji et al., 2018). Hybridized and advanced optimization techniques have also been tried to overcome the limitations with traditional AI techniques and to address the OASP problem effectively (Akpinar et al., 2013; Gunji et al., 2017; Bahubalendruni and Biswal, 2017; Sahoo, 2017; Bahubalendruni and Biswal, 2018a). Also, the hybrid heuristic approaches were implemented for multiple constraints in various realist problems like process planning and plant layout design (Bhunia et al., 2017; Bose and Pain, 2018; Turgay, 2018). Fine tuning the optimization process parameters hybrid algorithms is computationally inefficient. Moreover, the researchers only focused on linear assembly systems where subassembly identification is ignored. The recent research focused on stable subassembly identification for small products made with fewer parts (Bahubalendruni and Biswal, 2018b; Bahubalendruni and Kumar, 2018).

The proposed method aims to apply a new swarm intelligence based optimization algorithm, namely Elephant Swarm Water Search Algorithm (ESWSA), to solve subassembly based optimal assembly sequence planning efficiently for large scale products.

2. Assembly Attributes
Assembly attributes plays significant role in ensuring the practical possibility of assembly operation while applied during assembly sequence generation. Most of the assembly attribute data can be extracted through Computer Aided Design (CAD) product through user interface (Vigano and Osorio Gomez, 2012; Bahubalendruni and Biswal, 2016; Bahubalendruni and Biswal, 2017; Bahubalendruni and Biswal, 2018). Most of Application Program Interface (API) of CAD ease the process of automatic attribute extraction without any human error. A 17-part gear box assembly shown in Figure 1 is considered from literature.
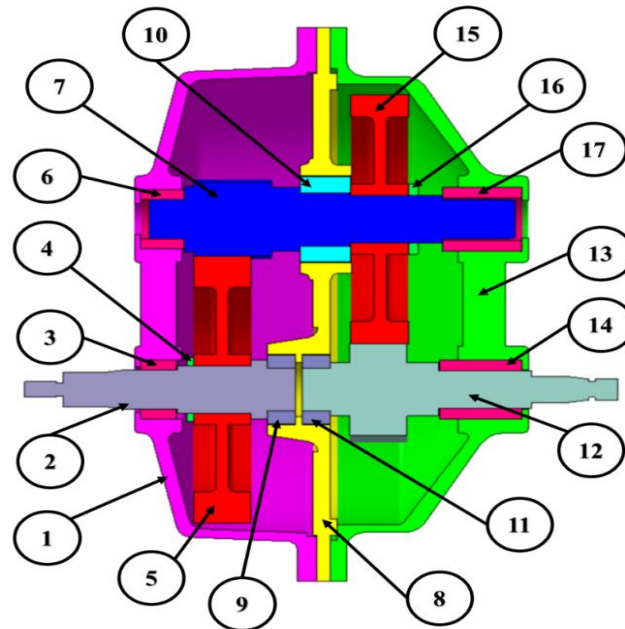
Figure 1. 17-part reduction gear box assembly cut section view

There are three assembly attributes namely Assembly Contact Matrix (ACM), Collision Free Path Matrix (CFPM) and Assembly Stability Matrix (ASM). ACM is a square matrix of size   equal to number of parts in the final product. It represents the surface contact between any pair of parts, the existence of surface contact is "1" and non-existence as "0". A product can have three CFPMs along principal axes (x+, y+ and z+) the transpose of these matrices give the CFPM along the opposite directions (x-, y- and z-), the element value "1" in the matrix represents possibility of collision free path of the part stated in the row in presence of part stated in the column. ASM is a square matrix of size   equal to number of parts in the final product. It characterizes the stability of any pair of parts, "3" signifies permanent stability between pair of parts due to external connectors, "2" represents permanent stability between pair of parts due to part mating features, "1" depicts partial stability of part stated in the column in presence of part in the row and "0" denotes unstable pair of parts. The ACM, ASM and CFPM for the reduction gear-box assembly is given in Figure 2.

```
        1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
 1 [    0  0  1  0  0  1  0  1  0  0  0  0  0  0  0  0  0 ]
 2 [    0  0  1  1  1  0  0  0  1  0  0  0  0  0  0  0  0 ]
 3 [    1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 ]
 4 [    0  1  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0 ]
 5 [    0  1  0  1  0  0  1  0  0  0  0  0  0  0  0  0  0 ]
 6 [    1  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0  0 ]
 7 [    0  0  0  0  1  1  0  0  0  1  0  0  0  0  1  1  1 ]
 8 [    1  0  0  0  0  0  0  0  1  1  1  0  1  0  0  0  0 ]
 9 [    0  1  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0 ]
10 [    0  0  0  0  0  0  1  1  0  0  0  0  0  0  1  0  0 ]
11 [    0  0  0  0  0  0  0  1  0  0  0  1  0  0  0  0  0 ]
12 [    0  0  0  0  0  0  0  0  0  0  1  0  0  1  1  0  0 ]
13 [    0  0  0  0  0  0  0  1  0  0  0  0  0  1  0  0  1 ]
14 [    0  0  0  0  0  0  0  0  0  0  0  1  1  0  0  0  0 ]
15 [    0  0  0  0  0  0  1  0  0  1  0  1  0  0  0  1  0 ]
16 [    0  0  0  0  0  0  1  0  0  0  0  0  0  0  1  0  0 ]
17 [    0  0  0  0  0  0  1  0  0  0  0  0  1  0  0  0  0 ]
```

(a) ACM of reduction gear box assembly

```
        1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
 1 [    0  0  2  0  0  2  0  0  0  0  0  0  3  0  0  0  0 ]
 2 [    0  0  2  2  2  0  0  0  2  0  0  0  0  0  0  0  0 ]
 3 [    2  2  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 ]
 4 [    0  2  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 ]
 5 [    0  2  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 ]
 6 [    2  0  0  0  0  0  2  0  0  0  0  0  0  0  0  0  0 ]
 7 [    0  0  0  0  0  2  0  0  0  2  0  0  0  0  2  2  2 ]
 8 [   -1  0  0  0  0  0  0  0  2  2  2  0 -1  0  0  0  0 ]
 9 [    0  2  0  0  0  0  0  2  0  0  0  0  0  0  0  0  0 ]
10 [    0  0  0  0  0  0  2  2  0  0  0  0  0  0  0  0  0 ]
11 [    0  0  0  0  0  0  0  2  0  0  0  2  0  0  0  0  0 ]
12 [    0  0  0  0  0  0  0  0  0  0  2  0  0  0  0  0  0 ]
13 [    3  0  0  0  0  0  0 -1  0  0  0  0  0  2  0  0  2 ]
14 [    0  0  0  0  0  0  0  0  0  0  0  0  2  0  0  0  0 ]
15 [    0  0  0  0  0  0  2  0  0  0  0  0  0  0  0  0  0 ]
16 [    0  0  0  0  0  0  2  0  0  0  0  0  0  0  0  0  0 ]
17 [    0  0  0  0  0  0  2  0  0  0  0  0  2  0  0  0  0 ]
```

(b) ASM of reduction gear box assembly

```
        1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
 1 [    0  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ]
 2 [    0  0  0  0  0  0  1  1  1  1  1  1  1  1  1  1  1 ]
 3 [    0  1  0  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ]
 4 [    0  1  0  0  1  1  1  1  1  1  1  1  1  1  1  1  1 ]
 5 [    0  1  0  0  0  1  1  1  1  1  1  1  1  1  1  1  1 ]
 6 [    0  1  1  1  1  0  1  1  1  1  1  1  1  1  1  1  1 ]
 7 [    0  1  1  1  1  0  0  1  1  1  1  1  1  1  1  1  1 ]
 8 [    0  0  0  0  0  0  1  1  0  0  1  1  1  1  1  1  1 ]
 9 [    0  0  0  0  0  0  1  1  1  0  1  1  1  1  1  1  1 ]
10 [    0  1  1  1  0  0  0  1  1  0  1  1  1  1  1  1  1 ]
11 [    0  0  0  0  0  0  1  1  0  0  1  0  1  1  1  1  1 ]
12 [    0  0  0  0  0  0  1  1  0  0  1  0  0  1  1  1  1 ]
13 [    0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 ]
14 [    0  0  0  0  0  0  1  1  0  0  1  0  1  0  1  1  1 ]
15 [    0  1  1  1  0  0  0  0  1  0  1  1  1  1  0  1  1 ]
16 [    0  1  1  1  1  0  0  1  1  0  1  1  1  1  0  0  1 ]
17 [    0  1  1  1  1  0  0  1  1  0  1  1  1  1  0  0  0 ]
```

(c) CFPM of reduction gear box assembly along “x+” direction

```
        1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
 1 [    0  0  0  0  0  0  0  0  0  0  0  0  1  1  1  1  1 ]
 2 [    0  0  0  0  0  1  1  0  0  1  1  1  1  1  1  1  1 ]
 3 [    0  0  0  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ]
 4 [    0  0  0  1  0  1  1  0  1  1  1  1  1  1  1  1  1 ]
 5 [    0  0  1  1  0  1  0  1  1  1  1  1  1  1  1  1  1 ]
 6 [    0  1  1  1  1  0  0  1  1  1  1  1  1  1  1  1  1 ]
 7 [    0  1  1  1  0  0  0  0  1  0  1  1  0  1  0  0  0 ]
 8 [    0  0  1  1  1  1  0  0  0  0  0  0  0  1  1  1  1 ]
 9 [    0  0  1  1  1  1  0  0  0  0  1  1  1  1  1  1  1 ]
10 [    0  1  1  1  1  1  0  0  1  0  1  1  0  1  1  1  1 ]
11 [    0  1  1  1  1  1  1  0  1  1  0  0  1  1  1  1  1 ]
12 [    0  1  1  1  1  1  1  0  1  1  0  0  0  0  0  1  1 ]
13 [    1  1  1  1  1  1  1  0  0  1  0  1  0  0  0  0  0 ]
14 [    1  1  1  1  1  1  1  0  1  1  1  1  0  0  0  1  1 ]
15 [    1  1  1  1  1  1  0  1  1  1  1  0  0  1  0  1  1 ]
16 [    1  1  1  1  1  1  0  1  1  1  1  1  0  1  1  0  1 ]
17 [    1  1  1  1  1  1  0  1  1  1  1  1  0  1  1  1  0 ]
```

(d) CFPM of reduction gear box assembly along “y+” direction

```
        1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
 1 [    0  0  0  0  0  0  0  0  0  0  0  0  1  1  1  1  1 ]
 2 [    0  0  0  0  0  0  0  0  0  1  1  1  1  1  1  1  1 ]
 3 [    0  0  0  1  1  0  0  1  1  1  1  1  1  1  1  1  1 ]
 4 [    0  0  1  0  1  1  0  1  1  1  1  1  1  1  1  1  1 ]
 5 [    0  0  1  1  0  1  0  1  1  1  1  1  1  1  1  1  1 ]
 6 [    0  1  1  1  1  0  0  1  1  1  1  1  1  1  1  1  1 ]
 7 [    0  1  1  1  1  0  0  0  1  0  1  1  0  1  0  0  0 ]
 8 [    0  0  1  1  1  1  0  0  0  0  0  0  0  1  1  1  1 ]
 9 [    0  0  1  1  1  1  0  0  0  1  1  1  1  1  1  1  1 ]
10 [    0  1  1  1  1  1  0  0  1  0  1  1  0  1  1  1  1 ]
11 [    0  1  1  1  1  1  0  0  1  0  0  0  1  1  1  1  1 ]
12 [    0  1  1  1  1  1  0  0  1  0  0  0  0  0  0  0  0 ]
13 [    1  1  1  1  1  1  0  0  1  0  1  0  0  0  0  0  0 ]
14 [    1  1  1  1  1  1  0  1  1  1  1  0  0  0  1  1  0 ]
15 [    1  1  1  1  1  1  0  1  1  1  1  1  0  1  0  1  1 ]
16 [    1  1  1  1  1  1  0  1  1  1  1  1  0  1  1  0  1 ]
17 [    1  1  1  1  1  1  0  1  1  1  1  1  0  1  1  1  0 ]
```

(f) CFPM of reduction gear box assembly along “z+” direction

Figure 2. (a) to (f) represents the reduction gear-box assembly in ACM, ASM and CFPM

3. Elephant Swarm Water Search Algorithm

Elephant Swarm Water Search Algorithm is a swarm based optimization algorithm inspired from the natural phenomenon of Elephants searching for water during dry weather and scarcity of water situations (Mandal, 2018). The communication strategies among swarm is used to find the global optimal solution from local optima. A typical ESWSA for assembly sequence planning is given in Figure 3.
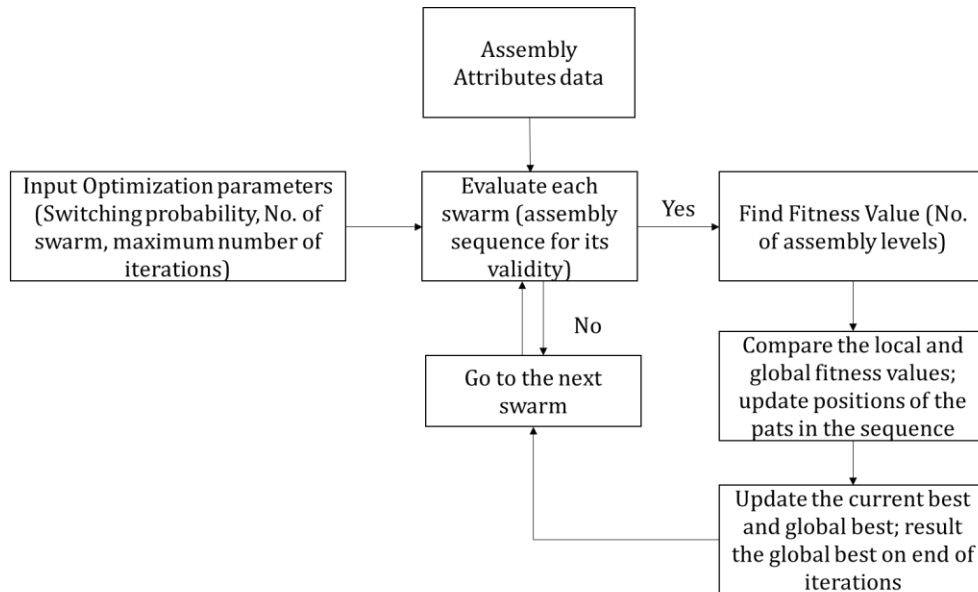

Figure 3. Flow diagram of ESWSA for assembly sequence generation

4. Assembly Sequence Validation and Fitness Value Computations

Initial elephants swarm are generated based on the random numbers; in each swarm, assembly-part numbers are assigned to random numbers in ascending order. A typical assembly sequence generation from random numbers is indicated in Table 1.

Table 1. Assembly sequence generation from random numbers

SWARM-1																
Random No.	0.98378	0.18833	0.27585	0.09189	0.91728	0.15165	0.17076	0.31490	0.37927	0.84244	0.95127	0.72750	0.01516	0.14875	0.19045	0.94416	0.83014
Assembly sequence	17	6	8	2	14	1	5	9	10	13	16	11	1	3	7	15	12
SWARM-2																
Random No.	0.17272	0.22271	0.68025	0.34256	0.67588	0.56382	0.54694	0.51218	0.72010	0.59573	0.72232	0.89518	0.88814	0.88244	0.09004	0.96072	0.87177
Assembly sequence	2	3	10	4	9	7	6	5	11	8	12	16	15	14	1	17	13

Liaison become true when each successive part in the sequence must exhibit at least one surface contact with one part in the preceding list. Assembly stability is true when part has at least one stable relation (partial/permanent) with reference to one part the preceding sequence. Geometric feasibility is true when part has at least one collision free path in the existence of all preceding parts. A stable subassembly is a sub-set where all the parts must exhibit permanent stability along with geometric feasibility. The position changes for the parts after every iteration based on the fitness values is given in Table 2.

Table 2. Assembly sequence validation criterion

Assembly Predicate	Predicate testing criterion	Remarks
Liaison Predicate	$\sum_{j=1}^{k} ACM(j,k) \geq 1; \quad \forall k \in [2,n]$	
Geometric Feasibility	$\sum_{j=1}^{k} CPFM_d(j,k) = k; \quad \forall k \in [2,n] \, \& \, \exists d \in [1,6]$	*d is a feasible direction*
Assembly stability	$\sum_{j=1}^{k} ASM(j,k) \geq 1; \quad \forall k \in [2,n]$	
Subassembly Detection	*If dj=dj+1; j ∈ (l,k) & k>l* *If ASM (j,j+1)>=2 j ∈ (l,k) & k>l*	The group of parts (l,k) creates a stable subassembly

The present problem part/subassembly position swapping in the assembly sequence obtained from above fitness vales are depicted in Figure 4.
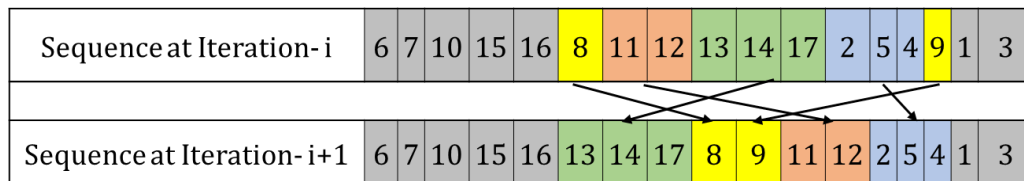

Figure 4. Part/Subassembly position swapping in the assembly sequence

Fitness function is evaluated based on the number of total assembly levels, which is calculated as the sum of maximum number of single assembly operations of the final assembly and sub-assembly. A typical assembly level calculation for a sequence is listed in the Table 3.

Table 3. Assembly level calculation for valid sequence

Assembly level	Assembly sequence																
L7	((((((6-7)-10)-15)-16)-((8-11)-12)-((13-14)-17)))-(((2-5)-4)-9)-(1-3))																
L6	(((((6-7)-10)-15)-16)-((8-11)-12)-((13-14)-17))																
L5	((((6-7)-10)-15)-16)-((8-11)-12)																
L4	(((6-7)-10)-15)-16											((2-5)-4)-9)-(1-3)					
L3	((6-7)-10)-15											((2-5)-4)-9					
L2	(6-7)-10					(8-11)-12			(13-14)-17			(2-5)-4					
L1	6-7					8-11			13-14			2-5				1-3	
Loose parts	6	7	10	15	16	8	11	12	13	14	17	2	5	4	9	1	3

The maximum number of iterations and population of algorithms (swarm) is chosen as 1000, 10 respectively and the probabilistic constant (switching probability) is turned to 0.86 after few iterations and the resulted outcomes are presented in the following section.

5. Results and Discussions
The product is solved multiple times due to the stochastic nature of optimization algorithm and the resulted assembly sequence is compared with that of solution given by past literature. Dini has proposed a solution while implementing the Genetic Algorithm (GA), the solution has 12 total assembly levels with 4 identified stable subassemblies (Dini and Santochi, 1992). Trigui solved the product with disassembly sequence planning approach and obtained a feasible solution with 7 assembly levels (Trigui et al., 2017). The proposed approach resulted in an assembly sequence with 6 total assembly levels as listed in Table 4.

Table 4. ESWSA resulted optimal assembly sequence with level description

Assembly level	Assembly sequence																
L6	((((((2-5)-4)-(1-3))-((8-9)-(11-12)))-(((((6-7)-10)-15)-16)-((13-14)-17)))																
L5									(((((6-7)-10)-15)-16)-((13-14)-17))								
L4	((((2-5)-4)-(1-3))-((8-9)-(11-12)))								((((6-7)-10)-15)-16								
L3	(((2-5)-4)-(1-3))								(((6-7)-10)-15								
L2	((2-5)-4)			((8-9)-(11-12))					((6-7)-10)					((13-14)-17)			
L1	(2-5)		(1-3)	(8-9)		(11-12)			(6-7)					(13-14)			
L0 (Loose parts)	2	5	4	1	3	8	9	11	12	6	7	10	15	16	13	14	17

The rate of convergence of the algorithm is compared with that of basic scheme of Genetic Algorithm and Particle Swarm Optimization (PSO) technique and presented in Figure 5. The time taken to converge the solution by each algorithm is presented in Figure 6. The GA and PSO have taken the more time than ESWSA.
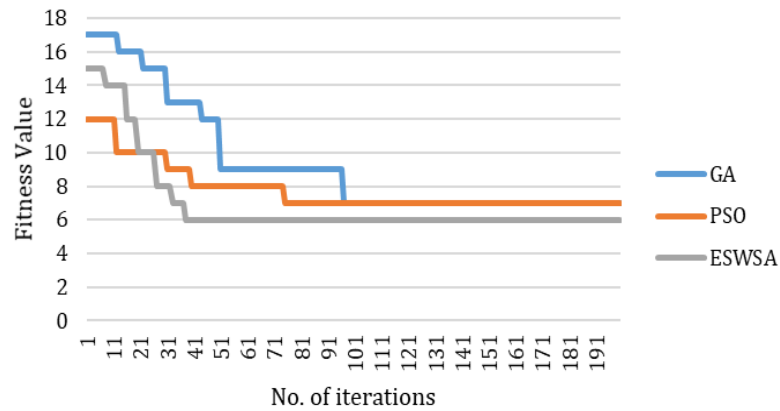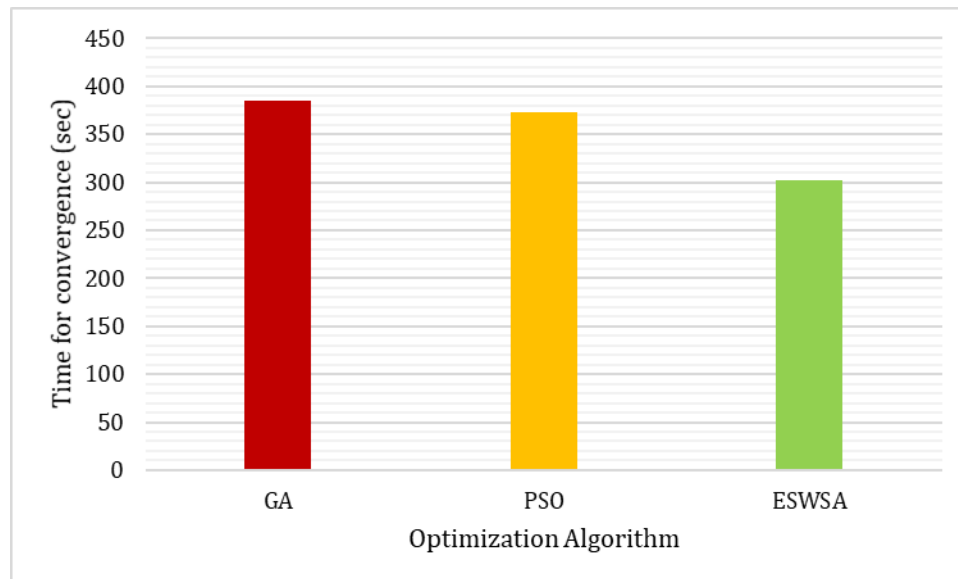
Figure 5. Convergence plot of ESWSA

Figure 6. Time of convergence

6. Conclusion

An efficient method to produce optimal assembly sequences by considering all the necessary assembly sequence validity constraints is proposed. The proposed ESWSA is a proven capable to solve both subassembly detection and assembly sequence planning. The previous methods discussed by Dini and Trigui focus on identifying the stable subassemblies and then each subassembly is treated as a single part and further linear assembly sequence is found out. Unlike the previous methods, the proposed method obtains the better fitness value with total number of assembly levels of 5, where subassembly detection is an integral part of assembly sequence generation. A comparative assessment is also presented with that of swarm based optimization technique to corroborate the performance of the proposed method.

Conflict of Interest
The authors declare that there is no conflict of interest for this publication.

References

Akpınar, S., Bayhan, G.M., & Baykasoglu, A. (2013). Hybridizing ant colony optimization via genetic algorithm for mixed-model assembly line balancing problem with sequence dependent setup times between tasks. *Applied Soft Computing*, *13*(1), 574-589.

Bahubalendruni, M.R., & Biswal, B.B. (2016a). A review on assembly sequence generation and its automation. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 230*(5), 824-838.

Bahubalendruni, M.R., & Biswal, B.B. (2016b). Liaison concatenation–a method to obtain feasible assembly sequences from 3D-CAD product. *Sadhana, 41*(1), 67-74.

Bahubalendruni, M.R., & Biswal, B.B. (2017). A novel concatenation method for generating optimal robotic assembly sequences. *Proceedings of the Institution of Mechanical Engineers. Part C: Journal of Mechanical Engineering Science, 231*(10), 1966-1977.

Bahubalendruni, M.R., & Biswal, B.B. (2018a). An intelligent approach towards optimal assembly sequence generation. *Proceedings of the Institution of Mechanical Engineers. Part C: Journal of Mechanical Engineering Science*, *232*(4), 531-541.

Bahubalendruni, M.R., & Biswal, B.B. (2018b). An efficient stable subassembly identification method towards assembly sequence generation. *National Academy Science Letters, 41*(2), 1-4.

Bahubalendruni, M.R., Biswal, B.B., & Khanolkar, G.R. (2015a). A review on graphical assembly sequence representation methods and their advancements. *Journal of Mechatronics and Automation, 1*(2), 16-26.

Bahubalendruni, M.R., Biswal, B.B., Kumar, M., & Nayak, R. (2015b). Influence of assembly predicate consideration on optimal assembly sequence generation. *Assembly Automation, 35*(4), 309-316.

Bahubalendruni, M.R., Deepak, B.B.V.L., & Biswal, B.B. (2016). An advanced immune based strategy to obtain an optimal feasible assembly sequence. *Assembly Automation, 36*(2), 127-137.

Bahubalendruni, M.R., & Kumar, G.A. (2018). Practically feasible optimal assembly sequence planning with tool accessibility. In *IOP Conference Series: Materials Science and Engineering, 390*(1), 12-26.

Bhunia, A.K, Duary, A., & Sahoo, L. (2017). A genetic algorithm based hybrid approach for reliability-redundancy optimization problem of a series system with multiple-choice. *International Journal of Mathematical, Engineering and Management Sciences*, *2*(3), 185-212.

Bose, G.K, & Pain, P. (2018). Metaheuristic approach of multi-objective optimization during EDM process. *International Journal of Mathematical, Engineering and Management Sciences*, *3*(3), 301-314.

Chen, W.C., Hsu, Y.Y., Hsieh, L.F., & Tai, P.H. (2010). A systematic optimization approach for assembly sequence planning using taguchi method, DOE, and BPNN. *Expert Systems with Applications, 37*(1), 716-726.

Deepak, B.B.V.L., Bala Murali, G., Bahubalendruni, M.R., & Biswal, B.B. (2018). Assembly sequence planning using soft computing methods: A review. *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering,* doi: 10.1177/0954408918764459.

Dini, G., & Santochi, M. (1992). Automated sequencing and subassembly detection in assembly planning. *CIRP Annals, 41*(1), 1-4.

Ghandi, S., & Masehian, E. (2015). A breakout local search (BLS) method for solving the assembly sequence planning problem. *Engineering Applications of Artificial Intelligence,* 39, 245-266.

Gunji, A.B., Deepak, B.B.B.V.L., Bahubalendruni, C. R., & Biswal, D. B. B. (2018). An optimal robotic assembly sequence planning by assembly subsets detection method using teaching learning-based optimization algorithm. *IEEE Transactions on Automation Science and Engineering*, *15*(3), 1369-1385.

Gunji, B., Deepak, B.B.V.L., Bahubalendruni, M.V.A.R., & Biswal, B. (2017). Hybridized genetic-immune based strategy to obtain optimal feasible assembly sequences. *International Journal of Industrial Engineering Computations*, *8*(3), 333-346.

Gunji, B.M, Deepak, B.B.V.L., Khamari, B.K., & Biswal, B.B. (2019). CAD-based automatic clash analysis for robotic assembly. *International Journal of Mathematical, Engineering and Management Sciences*, *4*(2), 432-441.

Mandal, S. (2018). Elephant swarm water search algorithm for global optimization. *Sadhana, 43*(1), 1-21.

Smith, S.S-F., Smith, G.C., & Liao, X. (2001). Automatic stable assembly sequence generation and evaluation. *Journal of Manufacturing Systems*, *20*(4), 225-235.

Sahoo, L. (2017). Genetic algorithm based approach for reliability redundancy allocation problems in fuzzy environment. *International Journal of Mathematical, Engineering and Management Sciences, 2*(4), 259–272.

Trigui, M., Belhadj, I., & Benamara, A. (2017). Disassembly plan approach based on subassembly concept. *The International Journal of Advanced Manufacturing Technology, 90*(1-4), 219-231.

Turgay, S. (2018). Multi objective simulated annealing approach for facility layout design. *International Journal of Mathematical, Engineering and Management Sciences, 3*(4), 365–380.

Vigano, R., & Osorio Gomez, G. (2012). Assembly planning with automated retrieval of assembly sequences from CAD model information. *Assembly Automation, 32*(4), 347-360.